

# Ingeniería de requerimientos, camino a un proyecto exitoso

## Requirements engineering, the road to a successful project

Contreras Bernal, Carlos Andrés <sup>1</sup> Hernández Ruiz, Catalina María <sup>2</sup>

### Citar este documento:

Contreras Bernal, Carlos Andrés. Hernández Ruiz, Catalina. Ingeniería de requerimientos, camino a un proyecto exitoso. Revista Technol.Investig.Academia TIA, ISSN: 2344-8288, 10 (2), pp. 90-115. Bogotá-Colombia.

---

<sup>1</sup> Ingeniero en control, Universidad Distrital Francisco José de Caldas, Líder de soporte segundo nivel, Bogotá Colombia, caacontrerasb@correo.udistrita.edu.co

<sup>2</sup> Ingeniera de Sistemas, Universidad Distrital Francisco José de Caldas, Senior Project Manager, Bogotá Colombia, cmhernandezr@correo.udistrita.edu.co

## Resumen

El presente artículo desarrolla una revisión sistemática de literatura enfocada en el contexto de la ingeniería de requerimientos, para la comprensión, análisis e identificación de metodologías, estándares y autores relevantes. Mediante la investigación y clasificación de referencias, se genera una revisión del tema que aborda desde los conceptos iniciales hasta los modelos actuales aceptados y aplicados, pasando por evolución histórica, definición de elementos y roles, así como un contraste de comunidad, organizaciones y estándares internacionales.

**Palabras Clave:** *Ingeniería de requerimientos, tipos de requerimientos, analista de requerimientos, interesados, fases de la Ingeniería de requerimientos, técnicas de modelado de requerimientos.*

### Abstract

This paper shows a systematic literature review focused on the context of requirements engineering. In order to understand, analyze and identify relevant methodologies, standards and authors. Through research and reference classification, a revision of the topic is generated from initial concepts to current applied models, passing through historical evolution, definition of elements and roles, as well as a contrast of community, organizations and international standards.

**Key Words:** *Requirements Engineering, Requirements types, Requirements Analyst, Stakeholders, Phases of Requirements Engineering, Requirements Modeling Techniques.*

## I. Introducción

Los atributos, características y objetivos de un sistema o de un producto deben ser analizados, especificados y gestionados para su transformación en soluciones tangibles o intangibles que respondan ante las necesidades y expectativas del cliente. Su definición, dirección y manejo han sido ampliamente discutidos desde hace más de 60 años, a partir del surgimiento de métodos, técnicas y metodologías que generaron procesos y herramientas para el modelamiento de los sistemas finales.

A lo largo de este documento se plantea una revisión sistemática de literatura sobre la ingeniería de requerimientos, abordando no solo definiciones iniciales y componente histórico, sino también contrastando las posturas de los autores más relevantes en esta subdisciplina, quienes construyeron metodologías enfocadas en productos de software, pero adaptables a cualquier tipo de industria, convirtiéndolas en marcos de trabajo referentes a nivel mundial.

El contenido está desarrollado de la siguiente manera: Se presenta en la sección 2 una contextualización del concepto de requerimiento, y sus posibles enfoques dependiendo del entorno al que aplique. En el apartado 3, se especifican los tipos de requerimientos y sus diferencias. Con este preámbulo se da lugar a las secciones 4 y 5 que describen la ingeniería de requerimientos y su progreso desde 1960, mostrando el desarrollo y evolución de técnicas y metodologías que consolidaron a esta subdisciplina de la ingeniería de sistemas como una etapa clave en la ejecución de proyectos. Los puntos 6 y 7 describen dos roles que juegan un papel esencial en el proceso, el primero es el del analista de requerimientos, y el segundo el de los interesados en el desarrollo del sistema o producto, los denominados Stakeholders. En la sección 7 no solo se abordarán sus conceptos, sino también su clasificación y relevancia para los proyectos de TI. El apartado 8 resume los principales estándares a nivel internacional en materia de requerimientos, los cuales están relacionados a la sección 9 que expone las posturas de los autores más relevantes y actuales sobre las fases o el proceso de ingeniería de requerimientos. El documento finaliza con las secciones 10 y 11 de conclusiones a partir del análisis de la investigación realizada y las propuestas de trabajos futuros, por último, el cierre con las referencias bibliográficas en la sección 12.

### **¿Qué es un requerimiento?**

No existe una definición universal para esta palabra, numerosos autores, así como organizaciones han dado su punto de vista a lo largo del tiempo, sin embargo, la mayoría concuerda en que las palabras necesidad, atributo y servicio deben hacer parte de dicha definición [1]. Así las cosas, para este artículo se presenta la definición de Sommerville [2]: *“son la descripción de los servicios proporcionados por el sistema y sus*

*restricciones operativas. Estos requerimientos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema” [2] [3].*

Los requerimientos enmarcan las necesidades y compromisos que son acordados con el cliente, los cuales deben ser verificables, concisos, completos y consistentes, no deben dar cabida a ambigüedades o múltiples interpretaciones [4]. Existen tres niveles de descripción para los requerimientos, los cuales son:

**Negocio:** Enmarca los objetivos de alto nivel que la organización o cliente necesita, y cuyo cumplimiento le dará valor al servicio. Permitiendo observar el contexto político, industrial, y comercial donde se desenvuelve la misma.

**Usuario:** Declaración en lenguaje natural apoyada en diagramas de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar. Esto se hace de esta forma para que todos los stakeholders del proyecto los puedan comprender.

**Sistema:** Describe con detalle las funciones, servicios, y restricciones operativas del sistema. Debe definir exactamente qué es lo que se va a implementar. Se puede decir que son versiones ampliadas de los requerimientos de nivel de usuario [1] [2].

### **Tipos de requerimientos**

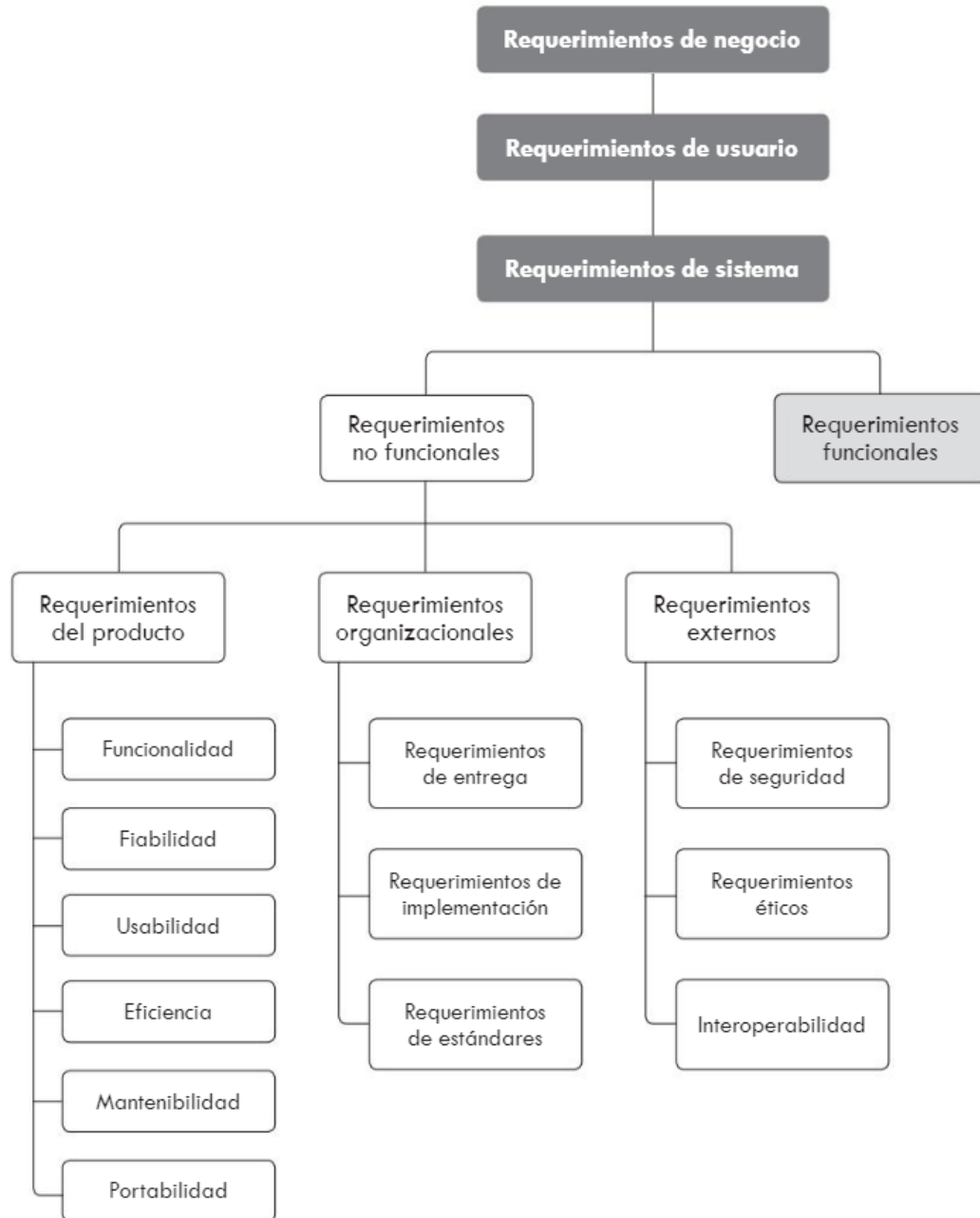
Con frecuencia se mencionan 2 grandes tipos de requerimientos, funcionales y no funcionales. Sin embargo, se recomienda tener cuidado con estas definiciones, puesto que no es raro encontrar que exista una relación entre las mismas más cercana de lo que se espera [5].

**Funcionales:** Describen con detalle lo que el sistema debe hacer, es decir, su comportamiento a partir de indicaciones sobre cuáles son las excepciones, entradas, y salidas del mismo. Por lo anterior dependen mucho del tipo de software, de los usuarios, así como de lo específico y particular que pueda ser el negocio de la organización.

**No Funcionales:** Son aquellos que no se refieren directamente a las funciones específicas, sino a los atributos de calidad inherentes como la fiabilidad, el tiempo de respuesta, la protección, la capacidad de almacenamiento, etcétera. Usualmente apuntan a alguno de los siguientes enfoques: Producto, Organizacional, o Externo.

Un desglose más detallado de los niveles y tipos de requerimientos puede ser observado en la ilustración 1.

*Ilustración 1. Niveles, tipos, y agrupación de requerimientos.*



*Fuente: [1]*

Algunos autores como Sommerville hacen referencia a una tercera tipificación de requerimientos, derivada del dominio de aplicación del sistema, refiriéndose a los requerimientos que se relacionan con estándares de

operación, definición de fórmulas matemáticas, entre otros, siempre y cuando restrinjan los requerimientos funcionales identificados, o que implique generar algunos nuevos [2] [6].

## **Ingeniería de requerimientos**

Con el auge de las metodologías ágiles se ha malentendido la importancia y el papel de los requerimientos, puesto que comúnmente el equipo de desarrollo inicia sus actividades sin tener el entendimiento claro de lo que deben hacer, y se escudan en que el mundo va tan rápido, y que la necesidad es tan inmediata que es mejor usar la experiencia de las primeras iteraciones de desarrollo para ir completando el entendimiento del negocio, y de la necesidad puntual del cliente [7].

La ingeniería de requerimientos es el conjunto de tareas y actividades que llevan a comprender claramente los requerimientos. Es una de las acciones más importantes dentro de la ingeniería de software, comenzando con la actividad de comunicación y estando presente durante la etapa de modelado. Teniendo en cuenta lo multidisciplinar que puede ser cada negocio, la ingeniería de requerimientos debe ser adaptativa según el proyecto al que se pretenda aplicar [2] [8]. Esta *“proporciona el mecanismo apropiado para entender lo que desea el cliente, analizar las necesidades, evaluar la factibilidad, negociar una solución razonable, especificar la solución sin ambigüedades, validar la especificación y administrar los requerimientos a medida que se transforman en un sistema funcional”* [7].

Aplicar la ingeniería de requerimientos en un proyecto implica gestionar la complejidad y la comunicación entre los diversos stakeholders. Teniendo en cuenta que el cambio y volatilidad pueden presentarse en cualquier momento, es necesaria una buena dirección y coordinación de actividades y procesos, en los que están inmersas personas y recursos. El valor de esta etapa en el ciclo de vida del software adquiere cada vez mayor peso y relevancia, ya que puede impactar a todo el desarrollo del proyecto [9].

Todo lo descrito anteriormente parece utópico, pero cuando se aplica consistentemente, resulta que facilita la medición y comprobación de los requisitos, aportando como consecuencia un producto de calidad y consistencia.

## **Historia y evolución**

A partir de 1960, se desarrollaron técnicas orientadas a analizar y transformar necesidades y objetivos de negocio en funcionalidades o componentes de un programa informático.

Los primeros métodos utilizaban un proceso de “divide y vencerás” para fragmentar el sistema en partes y de esta forma plantear las características, funciones y objetivos de cada una, que en suma serían los que debía cumplir en su totalidad. Posteriormente se buscó involucrar a los usuarios para agilizar los procesos de documentación y planteamiento de necesidades. Luego, la influencia de la ingeniería de software, del modelamiento de sistemas orientados a objetos, la creación de UML como estándar, entre otros, propiciaron la creación de la subdisciplina denominada Ingeniería de requerimientos [10] [11].

A continuación, algunas de las técnicas más relevantes de la época que dieron lugar a este conjunto de procesos, actividades y técnicas que poco a poco consolidaron a la ingeniería de requerimientos como uno de los elementos claves en la consecución de un proyecto.

## II. Análisis estructurado y técnica de diseño (SADT)

Uno de los primeros métodos fue el de Análisis estructurado y técnica de diseño (SADT), el cual se basa en diagramas de datos y de actividades, para expresar un diagnóstico tanto del sistema actual como del esperado, esta técnica se caracteriza por desarrollar la construcción de modelos para abordar sus diferentes fases así [12]:

**Modelo del Entorno Actual:** Permite el entendimiento de cómo está configurado y cómo funciona el sistema actual, buscando identificar cuáles son las partes que serán mejoradas, replanteadas o cambiadas por el nuevo software a desarrollar.

**Modelo de Operaciones Actuales:** Expresa como el sistema actual es utilizado por usuarios finales y demás stakeholders, nuevamente potencializando aquellos componentes que tienen oportunidades de mejora, o que evolucionaron en el nuevo producto.

**Modelo del Nuevo Sistema:** Este es un conjunto de modelos que describen las expectativas y objetivos del nuevo sistema:

- **Modelo de Requerimientos:** Lista los requerimientos esperados para el nuevo software.
- **Modelo Funcional:** Detalla y especifica los requerimientos funcionales del sistema.

- **Modelo de Diseño y de mecanismos:** Plantea el diseño global que cumple con todos los requerimientos y muestra cómo se relacionan los diferentes elementos del producto deseado, partiendo de la premisa “divide y conquistarás”.
- **Modelo del Plan de Test:** Diseña el plan de pruebas del nuevo sistema, determinando la satisfacción de los usuarios, y encontrando oportunidades de mejora para que el producto evolucione.
- **Nuevo Modelo del Entorno y de Operaciones:** Permite analizar cómo el nuevo software impactará en el entorno, qué variables nuevas podrá satisfacer y cómo generará beneficios a diversos stakeholders. Estos dos modelos son la siguiente versión de los realizados en el diagnóstico del sistema anterior, los nuevos responden al mismo análisis partiendo del uso y operación del producto mejorado o renovado.
- **Modelo de Conversión. y de Desarrollo del Sistema:** Detalla la puesta en marcha del sistema y el proceso de migración entre el anterior producto y el nuevo, especificando etapas del proyecto, roles y responsabilidades, y cronograma.

Como se puede observar, cada modelo desarrolla diferentes aspectos necesarios para la comprensión del sistema actual y del nuevo. Haciendo uso de diagramas de actividades y datos se pueden representar las actividades de un proceso, identificar sus relaciones, controles, mecanismos y dependencias, así como definir qué datos están inmersos en cada una [12] [13].

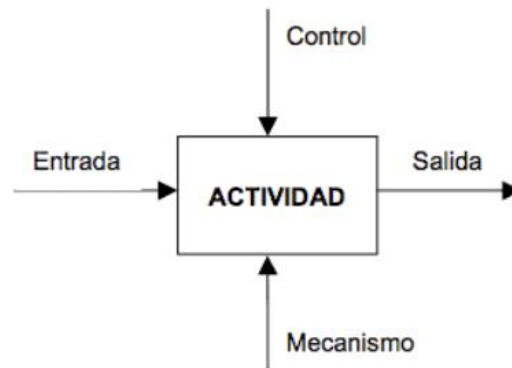
En los diagramas SADT son primordiales las actividades del proceso y los flujos que establecen la comunicación entre ellas. Las primeras se representan mediante una caja cuyo nombre corresponde a una función o parte activa del proceso, y los flujos mediante flechas pueden expresar alguno de los siguientes elementos [13].

- **Entrada:** Son aquellos insumos que se requieren para la realización de la actividad.
- **Salida:** Son el producto o resultados generados a partir de la transformación de las entradas a través de la ejecución de la actividad.
- **Control:** Son aquellas restricciones que impactan a la actividad y regulan la generación de salidas.
- **Mecanismo:** Son los usuarios que realizan la actividad, pueden ser recursos como personas, equipos u otros sistemas.



Para una representación visual de lo descrito anteriormente, es posible observar la ilustración 2.

*Ilustración 2. Elementos de un Diagrama de Actividades SADT*



*Fuente: [13]*

Las actividades en los diagramas SADT se ubican de forma organizada, de izquierda a derecha según la influencia que tienen sobre las demás; de esta forma las que pueden generar algún impacto, van a ser entradas o controles de las que son dependientes de ellas [13].

### **Análisis estructurado de sistema y método de diseño (SSADM)**

Es uno de los estándares más reconocidos en el mundo, su origen fue en el Reino Unido en 1980, como el anterior método, este plantea fases y establece los objetivos de cada una así [12]:

- **Análisis:** Documentar el modelo lógico del sistema actual, sus fortalezas y oportunidades para plantear los objetivos del nuevo sistema.
- **Especificación de requerimientos:** Busca detallar el modelo lógico del sistema deseado.
- **Selección de opción del sistema:** Busca identificar y especificar aquellos requerimientos que son opcionales para el nuevo software.
- **Diseño lógico de datos:** A partir de los requerimientos y del diseño lógico del modelo de la nueva aplicación, el siguiente paso es el modelamiento del flujo de los datos en la misma.
- **Diseño lógico de procesos:** Con el diseño general del software y de sus datos, se procede con la documentación de los procesos del sistema.
- **Diseño físico:** Se refiere al diseño o planteamiento de modelos para las tecnologías o plataformas que se utilizaran para generar el sistema [12] [14].

Así como en SADT se utilizan diagramas de datos y de actividades, en SSADM se aplican los modelos de entidad, diagramas de flujos de datos, matrices de entidades y funciones, historias de la vida de la entidad y los esquemas de proceso lógico [15].

## Sesiones JAD

En paralelo, en Estados Unidos, IBM desarrolla la técnica JAD, y adquiere tal éxito que es lanzada al público en los años 90, década en la que se vuelve reconocida a nivel mundial.

La técnica buscaba agilizar el proceso de modelado de requerimientos que planteaban las técnicas actuales, involucrando a los usuarios y fomentando los espacios de comunicación y de validación de las salidas y modelos. Este método se basa en sesiones o talleres para reunir, especificar y acordar los requerimientos del producto deseado, las cuales tienen las siguientes características:

- Agenda definida, buena preparación de la sesión y de larga duración.
- Sesiones productivas, en las que se construyen los modelos utilizando herramientas CASE para la generación de diagramas y demás documentación.
- Iterativas y correctivas, los modelos y diagramas deben ser evaluados constantemente por los participantes y demás stakeholders.
- Diversidad en los roles participantes para fomentar la discusión desde múltiples perspectivas. Roles y perfiles como: moderador, promotor, especialista en modelos y herramientas CASE, desarrolladores, usuarios, son relevantes en los espacios.
- Planeación y control. La preparación y ejecución de cada taller van acompañadas de las siguientes fases:
  - *Inicio*: Se definen los objetivos de la sesión, lugar, participantes, material y agenda.
  - *Desarrollo*: A partir de la interacción y comunicación se generan los modelos y posibles salidas del proyecto.
  - *Finalización*: Etapa en la que se validan los modelos propuestos para determinar las salidas, según los objetivos planteados [16] [17].

#### 5.4. Técnica de modelado de objetos (OMT)

Esta metodología fue creada en 1991 por Rumbaugh, Blaha, Premerlani, Eddy, y Lorensen, y es una de las más sólidas y maduras que se generaron en torno al análisis y diseño orientado a objetos. Define las siguientes fases:

- **Análisis:** Busca la construcción de un modelo que resuma los objetivos del sistema deseado, sus propiedades más importantes. No debe estar escrito en términos técnicos, ni detallar el cómo se construirá el software o aplicación, sino que debe poder ser entendido y discutido por expertos y demás stakeholders que puedan o no ser ajenos al componente tecnológico.
- **Diseño del sistema:** Esta fase tiene como objetivo el planteamiento de la arquitectura del sistema y la organización de sus componentes y subsistemas.
- **Diseño de objetos:** Se basa en el diseño del modelo de objetos a través del análisis del sistema, abordando detalles específicos de la implementación como las estructuras de datos y los algoritmos que van a construirse para el desarrollo de cada clase.

Se le considera la etapa principal de esta metodología, en la que se definen las entidades y sus interacciones para lograr las funcionalidades deseadas.

Son vitales los diagramas de clases y de casos o instancias, siendo los primeros una descripción y planteamiento de las clases que componen el sistema. Mientras que los diagramas de casos concretos, expresan la forma en que los objetos del sistema se relacionan. Estos diagramas cuentan con los siguientes elementos: clases, objetos, atributos, operaciones, relaciones, asociaciones, entre otros [18].

En resumen, esta fase determina la forma en que el diseño puede ser implementado en distintos lenguajes y tecnologías.

**Implementación:** Los modelos de clases y de objetos se convierten en código. En esta fase es vital considerar y tener en cuenta los principios y patrones de ingeniería de software para que el sistema desarrollado sea flexible y escalable [19].

## **El analista de requerimientos**

Dentro de las fábricas de software existen diversos grupos de trabajo (Arquitectura, DBA, Seguridad, QA, Infraestructura, Desarrolladores, Gestores de Proyectos, Líderes, etc.), y muchos de ellos deben ser transversales. Por ejemplo, dentro de la gestión de proyectos el rol del analista de requerimientos presenta una relación con dos aspectos diferentes, la primera enfocada hacia el negocio, y la segunda hacia el grupo de desarrollo, convirtiéndose de alguna forma en el medio de comunicación entre ambos ámbitos [20].

**Hacia el negocio:** Esto implica que la persona que desempeñe este rol debe contar con las competencias relacionadas al levantamiento de requerimientos, como por ejemplo lo son la construcción de diagramas UML, modelado de procesos de negocio (BPMN), el manejo de documentación entorno al negocio, etc.

**Hacia el grupo de desarrollo:** Implica que el analista cuente con la capacidad y habilidad de trasladar los resultados de su trabajo al grupo de desarrollo, apoyándose nuevamente en diagramas UML, participando en la construcción de un modelo de dominio y preferiblemente aplicando conocimientos básicos de arquitectura de sistemas [21].

Según J. Robertson [22], este es un rol con gran potencial y posibilidad de invención, ya que no solo comprende el problema del negocio, sino que también conoce sobre las tecnologías y herramientas disponibles, por lo que su posición permite fomentar la creatividad y la innovación para la creación de soluciones que puedan sobrepasar las expectativas.

En pocas palabras el analista de requerimientos debe modelar lo que se conoce, lo que se extrae, lo que se identifica, y debe aprovechar dicho modelo para sentar las bases sólidas de las siguientes fases del desarrollo de un producto de software [7].

## **Stakeholders**

En la ingeniería de requerimientos este término apareció por primera vez en la década de los 90, cuando se hizo evidente que usar términos como comprador, cliente o usuario era una forma muy específica de referirse a los interesados en un proyecto, y que esto a su vez dejaba por fuera a muchos otros actores. Frente a esta evidente necesidad de agrupar, surgió la palabra Stakeholder con la siguiente definición:

Es una persona u organización que influye en los requerimientos de un sistema o que se ve afectado por el mismo [23].

Frecuentemente, se utilizan roles en lugar de personas. Los roles típicos de los Stakeholders en un proyecto de software comúnmente son los presentados en la tabla 1:

*Tabla 1. Grupos existentes en un proyecto.*

Stakeholder	Definición
Cientes/Usuarios	Personas u organizaciones internas o externas a la organización ejecutante que usarán el producto, servicio o resultado del proyecto.
Patrocinador	Persona o grupo que proporciona los recursos financieros, en efectivo o en especie, para el proyecto.
Directores del portafolio	Normalmente son ejecutivos de la organización que actúan como un panel de selección de proyectos.
Directores del programa	Responsables de la gestión coordinada de proyectos relacionados, interactúan con los directores de cada proyecto, proporcionándoles apoyo y guía en proyectos individuales.
Oficina de dirección de proyectos (PMO)	Cuerpo o entidad que tiene varias responsabilidades asignadas con relación a la dirección centralizada de los proyectos.
Directores del proyecto	Personas que tienen a su cargo todos los aspectos del proyecto.
Equipo del proyecto	Grupo de personas lideradas por el director del proyecto, encargados de llevar a cabo el trabajo del proyecto.
Gerentes funcionales	Persona con gran experiencia que puede proporcionar servicios al proyecto.
Gerentes operacionales	Personas encargadas de investigación y desarrollo, diseño, fabricación, aprovisionamiento, pruebas o mantenimiento de productos en la organización.
Vendedores/ Socios de negocios	Compañías externas que celebran un contrato para proporcionar componentes o servicios para el proyecto.

*Fuente: [24]. Nota: Clasificación de los stakeholders con su correspondiente descripción.*

Con el objetivo de identificar a los stakeholders la literatura propone clasificar en dos grandes tipos según su relación con el proyecto:

- **Stakeholders Directos:** Grupo de involucrados directamente con el ciclo de vida del proyecto, son afectados, y tienen un interés fuerte en el éxito de este.

- **Stakeholders Indirectos:** Grupo que tiene un bajo interés en el proyecto, y muestra cierta preocupación por el éxito de este [24].

Existen acciones claves para que los diversos stakeholders tengan mayor comprensión y claridad sobre los requerimientos [25]:

- Espacios y actividades que faciliten la adquisición de conocimiento en relación al dominio del problema y contexto de la tecnología.
- Flujos de comunicación transparentes y accesibles que permitan el intercambio de información, y la coordinación de actividades necesarias para el desarrollo de los procesos, depuración de dudas y refinamiento de los objetivos.

Se recomienda que estos espacios y acciones sean iterativos a lo largo del proyecto, ya que se les considera una buena práctica, pues fomentan la comunicación fluida y las alineaciones pertinentes para un correcto entendimiento y manejo de los requerimientos.

Antes de hablar del análisis de requerimientos, es importante tener en cuenta que no basta solo con conocer el significado de los stakeholders, sino que se requiere identificar cuántos y cuáles tiene un proyecto en específico, así como la función, límites e interés en el mismo. Lo anterior con el objetivo de definir la importancia que estos tendrán a lo largo del ciclo de vida del proyecto.

### III. Estándares de Ingeniería de Requerimientos

En el tema de ingeniería de requerimientos a pesar de que no existe un solo camino, o una sola metodología para obtener el resultado final con la calidad de la que se ha hablado, tampoco es necesario inventar la rueda, puesto que existen varios marcos de trabajo que sirven de patrón, y que pueden ser tomados como referencia puesto que reúnen las características más representativas de esta área. Estos estándares determinan las normas y directrices para describir cómo debe ser la especificación de los requerimientos finales [26]. Algunos de los más representativos son:

- **European Space Agency (ESA):** Prácticas de software que deben aplicarse a cualquier sistema computarizado de dicha entidad. Este estándar propone 2 tipos de requerimientos, los del usuario (capacidad, y restricción), y los del software (Funcionales, ejecución, interfaz, operación, recursos, verificación, aceptación-testeo, documentación, seguridad, portabilidad, calidad, fiabilidad, y mantenibilidad). A su vez brinda un conglomerado de pautas para tener en cuenta con relación a la definición de requerimientos [27][28].
- **Institute of Electrical and Electronics Engineers (IEEE):** Este estándar concibe un requerimiento como cualquier condición o capacidad que debe estar presente en un sistema. Propone la siguiente agrupación para los requerimientos: Requisitos de adaptación, de interfaces de externos, funcionales, de ejecución, lógicos de la base de datos, de restricciones de diseño, y los relacionados con atributos del sistema. Al igual que ocurre con la **ESA**, la **IEEE** también define una serie de condiciones a tener en cuenta durante la definición de requerimientos [27] [29].
- **Reference Model of Open Distributed Processing (RM-ODP):** Estándar publicado por ISO/IEC que está enfocado al desarrollo de aplicaciones distribuidas y abiertas. En este no se encuentra una clasificación, o agrupación detallada de los tipos de requerimientos, sin embargo, sí indica que un sistema debe contar con 5 puntos de vista: Empresa, Información, Computación, Ingeniería, y Tecnología [27] [30].
- **Métrica V3:** Metodología enfocada a la planificación, desarrollo, y mantenimiento de sistemas de información. El objetivo de este estándar es estructurar un grupo de técnicas, y productos traducidos en tareas específicas garanticen la calidad del proyecto. Define cinco grupos de requerimientos a considerar, como lo son: Funcionales, de rendimiento, de seguridad, de implantación, y de disponibilidad del sistema [27] [31]. Es importante destacar que Métrica V3 en la actualidad ha perdido campo de aplicación, puesto que se considera que frena el desarrollo paralelo de productos, al contar con un enfoque en cascada.

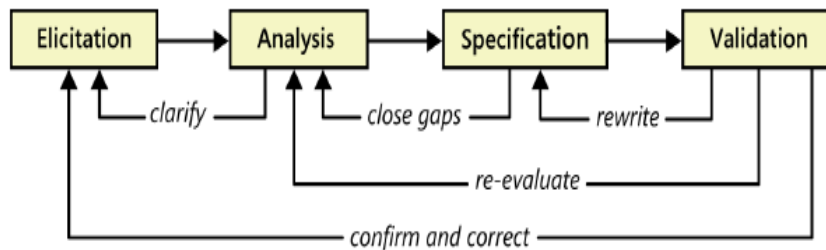
Adicional a los estándares, también existen algunas certificaciones internacionales como lo son la ISEB (Certificado en Ingeniería de Requisitos) e IREB (Certificado Profesional para Ingeniería de Requisitos) de la ISQI (Instituto Internacional de Calidad de Software) [32].

## Fases de Ingeniería de Requerimientos

Como se describió en el capítulo anterior, no existe un solo camino para desarrollar el proceso de ingeniería de requerimientos, sin embargo, existen fuentes ampliamente experimentadas que describen los pasos base que permiten adelantar correctamente este proceso a partir del éxito de sus trabajos.

Por un lado está Pressman [7] con la descripción de las siguientes siete tareas: (concepción, indagación, elaboración, negociación, especificación, validación y administración); la IREB [6] menciona cuatro macro-actividades: (elicitación, documentación, validación/negociación y gestión ); Sommerville [2] habla de cuatro subprocesos (estudio de viabilidad, obtención/análisis, especificación y validación); Wiegers [33] enuncia cuatro actividades generales: (elicitación, análisis, especificación y validación). Es claro que la visión de las fuentes expuestas es la misma, en la que se transita por un flujo que comienza por un entendimiento y reconocimiento del área de impacto del proyecto, se extrae y registra los requerimientos de diferentes formas, y se verifican y aprueban con el cliente, siempre realizando iteraciones entre todos estos procesos, para garantizar la calidad del proceso. Teniendo claro esto, se compartirá en mayor detalle la propuesta de Wiegers reflejada en la ilustración 3.

*Ilustración 3. Actividades ingeniería de requerimientos.*



*Fuente:[33]*

**Elicitación:** Consiste en una actividad humana donde se establecen las relaciones entre los stakeholders. Al ser la primera parte del proceso es quizás “la piedra angular” [34] del mismo, puesto que la calidad del producto dependerá de qué tan cercana a la perfección se desarrolle esta actividad.

Visto desde otro punto de vista, siempre que se quiera resolver un problema, lo primero que se debe hacer es averiguar más sobre el mismo [35]. Dependiendo de si se trata de un proyecto para un nuevo sistema, o para actualizar uno ya existente, se tendrán unas actividades ligeramente diferentes. Cuando se trata de actualizar sistemas existentes, las actividades son [36]:



- Reconocimiento y familiarización con el sistema antiguo.
- Entrevistas con los stakeholders.

Por otra parte, cuando se trata de la creación de un sistema nuevo las actividades son [36]:

- Un cuestionario específicamente elaborado para el caso de estudio, cuando ambas partes tienen claros los requerimientos.
- Compilación de necesidades y reuniones de grupo, cuando los clientes conocen los requerimientos, pero el grupo de desarrollo no.
- Generación iterativa de mockups a partir de ideas iniciales de requerimientos expuestas en reuniones de discusión, cuando los desarrolladores conocen los requerimientos, pero los clientes no.
- Aplicar métodos de investigación o de escenarios en donde se identifiquen los actores, logrando extraer las funcionalidades con mayor claridad e incluso llegando a definir historias de usuario, cuando ambas partes desconocen los requerimientos.

Todas estas actividades están encaminadas a comprender y experimentar las necesidades del cliente, y a su vez poderlas describir con éxito, sin olvidar nunca que los usuarios son la mayor fuente de requerimientos [37] [38] [39].

**Análisis:** Luego de esa etapa de elicitación, se requiere filtrar, pulir, refinar los requerimientos, para garantizar que todos los stakeholders los comprendan y analizarlos en busca de diferencias, errores u omisiones. Esta actividad implica hacer una definición más detallada de los requerimientos, así como una evaluación de su viabilidad y una priorización de los mismos [33].

Como se pudo observar en la definición de stakeholders, cada uno tiene una visión e interés diferente del proyecto. Debido a ello, durante esta fase es importante presentar los requerimientos con diferentes lenguajes, algunos preferirán el texto, pero para otros será más fácil de digerir con una imagen, lo importante es que los stakeholders lleguen a un entendimiento en común desde su visión particular del proyecto [36]. Las actividades de esta fase comúnmente son [40][41]:

- Reducir ambigüedades en los requerimientos.
- Analizar la factibilidad.
- Negociación de priorización.
- Aproximar los requerimientos a un lenguaje técnico.
- Plantear un modelo lógico.
- Analizar la interacción con otros sistemas.
- Documentar toda la fase de análisis

Cabe resaltar que, si los resultados de alguna de estas actividades no son los esperados, o si surgen dificultades, la mejor forma de resolverlas será haciendo uso de la fase anterior, iterando nuevamente las actividades de elicitación, y desarrollando una vez más las propias del análisis.

**Especificación:** En esta fase se aborda la documentación formal de los requerimientos, usando diferentes artefactos a partir de la tipología de estos. Se busca que dicha documentación sea coherente, accesible, revisable, y fácilmente comprensible por cada uno de los stakeholders [33]. Es recomendable en esta fase hacer uso de frases completas que definen la condición que causa que el sistema ejecute un comportamiento específico, además presentar la identificación clara de los actores, enfatizar la información más relevante, y evitar el uso de términos que sean ambiguos. Todo lo anterior apoyado por tablas, listas, figuras e imágenes [42]. Los pasos que se recomiendan seguir para esta fase son [40].

- Determinar los tipos de requerimientos.
- Elegir los artefactos a usar, procurando apoyarse en plantillas previamente establecidas.
- Representar los requerimientos.
- Documentar toda la fase de especificación.

Al igual que ocurre con la fase de análisis, puede que sea necesario regresar a la fase anterior, e incluso a la inicial, dependiendo de las dificultades o situaciones que surjan durante la ejecución de las actividades de especificación.

**Validación:** Con esta palabra se puede intuir que el objetivo de esta fase está centrado en garantizar que los requisitos sean correctos, tengan calidad, y satisfagan las necesidades del cliente. Lo anterior se debe a que estos requerimientos deben servir como una base confiable para la etapa de diseño, e incluso apoyar las pruebas finales del sistema y las de aceptación del usuario [33].

- Definir los criterios de aceptación.
- Revisar los requerimientos.
- Probar los requerimientos.
- Simular los requerimientos.

Finalmente es importante resaltar que las cuatro fases descritas en este artículo para el proceso de ingeniería de requerimientos son desarrolladas en paralelo con actividades específicas de administración que soportan la correcta continuidad de las fases [33].

- Control de cambios para los requerimientos.
- Análisis del impacto de control de cambios.
- Llevar un control de versiones, que permita revisar el histórico de los mismos.
- Llevar un seguimiento del estado de cada requerimiento, que facilite la identificación de problemas.
- Usar artefactos apropiados como matrices para la trazabilidad de los requerimientos. Se aconseja usar una herramienta de gestión de requisitos.

## **Estudios Relacionados**

Luego de esta revisión sistemática se considera pertinente presentar algunos casos de estudio en el ámbito local. Para el año 2016 A. Toro y L.E. Peláez [43], dos profesionales de las áreas de las TIC realizaron una tesis doctoral en la cual identificaron que los requerimientos incompletos son un factor clave que contribuye con mucha frecuencia al fracaso de un proyecto. Según lo indicado por la Asociación Colombiana de Ingenieros de Sistemas, [43] el 70 % de los proyectos en la industria de software no son exitosos. Por otra parte, observaron que el tiempo que se dedica a la fase de elicitación de requerimientos es inferior al 20%, sumado a la desconexión entre los conocimientos teóricos, y la experiencia real, que lleva a evidenciar que las instituciones de formación superior generan profesionales con conocimientos muy superficiales, mientras que los procesos de negocio demandan talento humano especializado en estas áreas.

Estos autores identificaron una situación llamativa en la región de Pereira - Colombia, puesto que a pesar de todo lo expuesto por la ingeniería de requerimientos, las Mipymes confían más en la experiencia acumulada que en marcos de trabajo altamente soportados para capturar la necesidad del usuario [43]. También

encontraron que para la fase de especificación muchas técnicas son conocidas por los colaboradores, pero con un nivel del 23,5% de aplicación, lo cual consideraron como bajo [43].

Finalmente A. Toro y L.E. Peláez [43] extraen tres grandes ideas de dicho estudio, las cuales son apoyadas por los autores de este artículo, y son:

- La industria local no ha abordado apropiadamente la producción de software, dejando a un lado la adopción de modelos enfocados a la calidad de la misma, y trayendo como consecuencia niveles bajos de satisfacción del cliente.
- Los profesionales no deben quedarse con los conocimientos adquiridos en las carreras de software, si no buscar las áreas específicas que busca la industria y especializarse en ellas.
- Es fundamental que la academia reciba una retroalimentación por parte de la industria de software, para ajustar sus procesos de formación, y así garantizar mejores herramientas para los futuros profesionales de estas áreas.

Otro estudio realizado por Zapata, Collazos, Giraldo y Sevilla [44] contribuye al análisis a partir de la aplicación de la ingeniería de requerimientos. Estos investigadores, desarrollaron un caso de estudio en el que se analizó la efectividad de aplicación de 3 técnicas de elicitación de requerimientos en equipos distribuidos y colocados. Dichos autores, plantearon un experimento que involucró a más de 30 estudiantes, y al menos 10 profesores de las Universidades del Cauca y del Quindío en Colombia, así como la de San Juan en Argentina.

El enfoque de su investigación se centró en la diferenciación entre los escenarios de elicitación de requerimientos para equipos que se encontraban en el mismo lugar versus aquellos que no, por lo que las aplicaciones y herramientas de comunicación fueron uno de los puntos clave a analizar.

Para el ejercicio se establecieron equipos de dos personas, que aplicaron una de las siguientes técnicas de elicitación: entrevista, entrevista-cuestionario y entrevista-brainstorming. Los estudiantes tuvieron el rol de ingenieros de requerimientos y los profesores de clientes. El contexto del proyecto fue la generación de un producto de software orientado a la gestión de información administrativa para la venta de tiquetes de viaje de una empresa de transporte.

Los investigadores que dirigieron el proyecto generaron un documento de requerimientos básicos que contenía todos los necesarios para la especificación del sistema solicitado, este se utilizó como referencia y base para contraste de las propuestas desarrolladas por los estudiantes.

También, para la revisión y análisis de la calidad de los requerimientos generados por los diferentes equipos, se establecieron algunas métricas de evaluación, como una calificación para la organización, asertividad del documento; el porcentaje de requerimientos sin defectos, ambigüedades o imprecisiones, y la relación entre los requerimientos propuestos versus los que se encuentran en el documento general de base que suministraron los investigadores. Estos aspectos permitieron calificar y evaluar las propuestas para así generar conclusiones y análisis.

Después de la ejecución de los ejercicios con cada equipo, los autores encontraron mayor efectividad en técnicas combinadas como las de entrevista-cuestionario, siendo un 10% más efectivas que las tradicionales que se definen en la literatura de ingeniería de requerimientos.

Así mismo, los investigadores realizaron encuestas para conocer aspectos relacionados a la experiencia en la aplicación de las técnicas de elicitación, como por ejemplo la usabilidad, siendo las entrevistas la herramienta con más intención y facilidad de uso.

Además de estas observaciones y análisis, Zapata, Collazos, Giraldo y Sevilla [44], en su investigación manifiestan que existen pocos estudios de comparación y valoración de las técnicas de ingeniería de requerimientos para evaluar su potencial y uso, más aún si está enfocado a equipos distribuidos.

#### IV Conclusiones

A lo largo de esta revisión sistemática de literatura se han contrastado diferentes metodologías y autores que facilitaron y fomentaron la evolución de la ingeniería de requerimientos. Desde los primeros pasos en los que el análisis estructurado fue clave, pasando por el enfoque orientado a objetos, la influencia de la ingeniería de software y la definición de lenguajes de modelado, emergen estándares y organizaciones internacionales que los regulan y documentan, para obtener lo que hoy en día se denomina una subdisciplina de la ingeniería de sistemas, y que se le considera una fase esencial para la ejecución de cualquier proyecto.

En relación con la pureza y base del requerimiento es vital asegurar que sea claro, conciso y específico, debe evitar cualquier ambigüedad para no generar confusiones o malas interpretaciones por los equipos o cualquier

stakeholder. Su construcción debe responder a la solución del “qué” y el “cómo” de tal forma que esté alineado con las necesidades y expectativas que se tienen en relación al producto o sistema. Debido a ello, el analista de requerimientos desempeña un papel fundamental, pues es el puente necesario entre los clientes/usuarios y el equipo de desarrollo. También, al tener el contexto de negocio y el del componente técnico es un rol con alto potencial creativo e innovador, ya que puede acercar los objetivos y deseos a modelos y diagramas que integren ambos entornos.

La ingeniería de requerimientos es un proceso fuertemente respaldado con más de 50 años de estudio y experiencia. Esto gracias al trabajo de diferentes entidades, y organizaciones involucradas en el mundo TIC. Define una serie de etapas para el desarrollo de procesos, identificando roles y actividades clave que se enmarcan en los estándares y guías internacionales. Su misión es lograr que cada uno de los stakeholders comprenda claramente los requerimientos, y se de una correcta gestión y desarrollo de estos. Por consiguiente, el conocimiento, definición, y caracterización de los interesados, permite establecer las relaciones, y facilita la recolección de información durante el proceso.

Sea que se incurriese por primera vez en la ingeniería de requerimientos, o que se tengan conocimientos previos, es aconsejable investigar y consultar a las instituciones que aportan estándares o certificaciones, puesto que son fuentes sólidas y fiables de este conocimiento. Es necesario tener en cuenta que la puesta en práctica de las fases no requiere de una ejecución lineal, y que si se aplican en proyectos con metodologías ágiles no es aconsejable ni recomendable que se salten u omitan.

Por último, no se puede dejar de resaltar la relevancia que tiene la fase de elicitación, ya que se le considera como la clave en la ingeniería de requerimientos. Al ser el punto de partida, esta representa el periodo en el que se tiene mayor acercamiento y entendimiento a las necesidades del usuario, por lo que debe buscar la más alta calidad para que el producto final también la tenga.

## V. Trabajo Futuro

A través del entendimiento y marco de referencia que ha permitido esta revisión de tema, se propone como trabajo futuro la puesta en práctica de un caso de estudio en el que se identifiquen y analicen los modelos de Pressman, Sommerville, y Wiegers, con el fin de contrastar y describir sus fases y elementos.

Las actividades que se desarrollan en la etapa de licitación permiten profundizar y experimentar con los atributos, características y necesidades que los usuarios expresan, estas brindan la oportunidad de conocer de

la mano de los diversos stakeholders las expectativas y metas del sistema o producto. Aplicando las técnicas de elicitación que los autores mencionan, se desea analizar sus ventajas y desventajas mediante diversos ejercicios en el entorno empresarial. Debido a la relevancia de esta fase, es vital profundizar en aquellas herramientas que pueden enriquecer la base de conocimiento y contexto asociado al proyecto y sus requerimientos.

Este nuevo caso de estudio permitirá examinar y relacionar las teorías aprendidas en esta revisión sistemática de literatura, identificando cuáles técnicas e instrumentos se articulan mejor a la cultura empresarial, proyectos, equipos y dinámica de la compañía.

Posterior a dicho análisis, de comparación y evaluación, se plantea identificar oportunidades de mejora para implantar o renovar mecanismos de elicitación de requerimientos en el departamento o empresa, según las características de los equipos, áreas de negocio y tipos de producto. De esta forma será viable proponer ya sea nuevos o mejoras para procesos actuales, incluyendo actividades, roles, formatos, y escenarios con el fin de eficientizar y acercar más a los usuarios e interesados en el desarrollo de los productos.

## VI. Referencias bibliográficas

- [1] M. P. Amórtegui, et al. Ingeniería de requerimientos. [En línea]. Disponible en: <http://hdl.handle.net/10554/44699>.
- [2] I, Sommerville. Ingeniería del software, Séptima Edición. Madrid (España). Pearson Educación S.A. 2005. 712 p.
- [3] M. Cristi. Introducción a la Ingeniería de Requerimientos. Centro Internacional Franco-Argentino de Ciencias de la Información y de Sistemas (CIFASIS). Argentina. 2011.
- [4] M. Arias. La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. Revista InterSedes © Universidad de Costa Rica. 2007.
- [5] M. Dabbagh, S. Lee, R. Parizi. Functional and non-functional requirements prioritization: empirical evaluation of IPA, AHP-based, and HAM-based approaches. Springer. USA. 2015.
- [6] P, Cuervo, D, Hernández. “Metodología para la Optimización de los Procesos de Recolección de Información y Análisis en la Etapa de Especificación de Requerimientos de Software,” 2017. [Online]. Available: <http://hdl.handle.net/11349/7804>
- [7] R.S, Pressman, *Ingeniería del software un enfoque práctico*. Séptima Edición. México. The McGraw-Hill Companies. 2010. 777 p

- [8] M.L, Perez. “Ingeniería de requerimientos”. Monografía . Instituto de ciencias básicas e ingenierías. Universidad autónoma del estado de Hidalgo. Pachuca. México. 2005.
- [9] D. Damian. Challenges in Requirements Engineering. University of Calgary. Canadá. 1999.
- [10] L. Capretz. A Brief History of the Object-Oriented Approach. ACM SIGSOFT Software Engineering Notes. Canadá. 2003.
- [11] A. Dennis, B. Wixom, D. Tegarden. System analysis and design. An object oriented approach with UML. Fifth edition. USA. 2015.
- [12] A. Amescua, Diseño de una metodología Case de desarrollo de software. Universidad Politécnica de Madrid. España, 1990.
- [13] F. Ahmed, S. Robinson, A. Tako, Using the structured analysis and design technique (SADT) in simulation conceptual modeling. Proceedings of the 2014 Winter Simulation Conference, Loughborough, 2014.
- [14] D. Ross, K. Schoman. Structured Analysis for Requirements Definition. IEEE Transactions on software engineering. 1977.
- [15] L. Macaulay, Requirements for Requirements Engineering Techniques. The University of Manchester, England. 1996.
- [16] E. Carmel, R. Whitaker, J. George. PD and Joint Application Design: A Transatlantic Comparison. Article in Communications of the ACM. 1993.
- [17] E. Davidson. Joint application design (JAD) in practice. Journal of Systems and Software. 1999.
- [19] S. Clyde. Notes on Object Oriented modeling and design. Brigham Young University. Utah. 1995.
- [18] R. Bourdeau, B. Cheng. A formal semantics for object model diagrams. IEEE Transactions on Software Engineering. 1995.
- [19] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Lorensen. Object-Oriented Modeling and Design. Prentice-Hall International. 1991.
- [20] R. Klendauer, M. Berkovich, R. Gelvin, J. Leimeister, H. Krcmar. Towards a competency model for requirements analysts. 2012
- [21] G. Pantaleo. Calidad en el desarrollo de software. Buenos Aires. Alfaomega. 2011. 208 p
- [22] J. Robertson. Requirements analysts must also be inventors. IEEE Software Journal. USA. 2005



- [23] Glinz, M., y Wieringa. R. J., Stakeholders in Requirements Engineering, IEEE, Software, Vol 24, No. 2, 2007.
- [24] Martínez Guerrero, J. M., y Silva Delgado, C. A. (2010, Noviembre 28). *Guía Metodológica para el levantamiento y análisis de Requerimientos de Software o Hardware en base a Procesos de Negocio*. Anexo 2. Universidad Javeriana. [En línea]. disponible en: [http://ual.dyndns.org/biblioteca/Evaluacion\\_Seleccion\\_Equipo\\_2017/pdf/S1d2.pdf](http://ual.dyndns.org/biblioteca/Evaluacion_Seleccion_Equipo_2017/pdf/S1d2.pdf)
- [25] D. Damian. Stakeholders in Global Requirements Engineering: Lessons Learned from Practice. IEEE Journal. Engineering, Computer Science. 2007.
- [26] K. Pohl. The Three Dimensions of Requirements Engineering. Seminal Contributions to Information Systems Engineering. Springer. Berlin. 2013.
- [27] A. Ramírez Fernández Tutora and A. Fraga Vázquez, “CLASIFICACIONES DE TIPOS DE REQUISITOS PARA LA MEJORA DEL PROCESO DE DESARROLLO DEL SOFTWARE” .2012.[Online]. Available: <http://hdl.handle.net/10016/14998>
- [28] M. Jones and U. Mortensen.. Guide to the software engineering standards. Francia. ESA Board for Software Standardisation and Control (BSSC). 1995. 6 p
- [29] G, Mendez. “Especificación de Requisitos según el estándar de IEEE 830”. Facultad de Informática - Universidad Complutense de Madrid. Octubre 2008. [Online]. Available: <https://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf>
- [30] Rm-odp.net. Febrero 2013. [Accedido 30 de Marzo de 2021]. [online] Available: <http://www.rm-odp.net/>
- [31] [ingsoftunne.wixsite.com](http://ingsoftunne.wixsite.com). Abril 2018. [Accedido 30 de Marzo de 2021]. [Online] Available: <https://ingsoftunne.wixsite.com/ingsoftunne/metodologia-metrica-v3>
- [32] L.E, Sanchez. “Papel de las certificaciones profesionales en la enseñanza universitaria de ingeniería de software en España”. Revista Española de Innovación, Calidad e Ingeniería del Software. Vol.6. No. 2. 2010. 24p
- [33] K, Wiegers. “Software Requirements”. Tercera Edición . Redmond. Washington. Microsoft Press. 2013. 637 p
- [34] M, Manies & U Nikual. “LA ELICITACIÓN DE REQUISITOS EN EL CONTEXTO DE UN PROYECTO SOFTWARE”. Ingenierías USBMed. Vol 2. No. 2. 2011. 5p
- [35] P. Loucopoulos and V. Karakostas, Systems Requirements Engineering. McGraw-Hill, 1995
- [36] P. Bourque and R. E. Fairley, SWEBOK. Guide to the Software Engineering Body of Knowledge. Version 3.0, 3rd. ed. IEEE Computer Society, 2014

- [37] I. Alexander and R. Stevens, Writing better Requirements. Pearson, 2002.
- [38] G. Gil. Herramienta para implementar LEL y escenarios (TILS). Universidad Nacional de La Plata. Argentina. 2002.
- [39] P.. Laplante. Requirements Engineering for Software and Systems. CRC Press. Tercera Edición. 2018.
- [40] M. Griselda, S.Barba. “Metodología DoRCU para la Ingeniería de Requerimientos”. IV Workshop en Requisitos. Argentina. 2001.
- [41] A. N. Camacho Herramienta para el análisis de requerimientos dentro de la pequeña empresa desarrolladora de software en Bogotá. [online]. Disponible en: <http://hdl.handle.net/10554/7480>.
- [42] M. Izaurralde. “Caracterización de Especificación de Requerimientos en entornos Ágiles: Historias de Usuario”. Tesis. Facultad Regional Córdoba. Universidad Tecnológica Nacional. Córdoba. Argentina. 2013.
- [43] Toro, A., & Peláez, L.E.. “*Ingeniería de Requisitos: de la especificación de requisitos de software al aseguramiento de la calidad. Cómo lo hacen las Mipymes desarrolladoras de software de la ciudad de Pereira.*” Entre Ciencia e Ingeniería. V. 10. No 20. 2016. 117-123.
- [44] S. Zapata, C. Collazos, E. Torres, F. Giraldo, G. Sevilla. Distributed Elicitation of Software Requirements: an experimental case from Argentina and Colombia. IEEE Journal. Colombian Conference on Computing (CCC). 2014.

Publicación Facultad de Ingeniería y Red de Investigaciones de Tecnología Avanzada – RITA

**REVISTA**

**TIA**